Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)

A Probabilistic Soft Logic Based Approach to Exploiting Latent and Global Information in Event Classification

Shulin Liu, Kang Liu, Shizhu He, Jun Zhao
National Laboratory of Pattern Recognition (NLPR)
Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China
{shulin.liu, kliu, shizhu.he, jzhao}@nlpr.ia.ac.cn

Abstract

Global information such as event-event association, and latent local information such as fine-grained entity types[1], are crucial to event classification. However, existing methods typically focus on sophisticated local features such as part-of-speech tags, either fully or partially ignoring the aforementioned information. By contrast, this paper focuses on fully employing them for event classification. We notice that it is difficult to encode some global information such as event-event association for previous methods. To resolve this problem, we propose a feasible approach which encodes global information in the form of logic using Probabilistic Soft Logic model. Experimental results show that, our proposed approach advances state-of-the-art methods, and achieves the best $F1$ score to date on the ACE data set.

Introduction

In the ACE (Automatic Context Extraction) event extraction task, an event is represented as a structure which is composed of a trigger, an event type, and the corresponding arguments with different roles. The objective of event extraction is to extract event instances of specific types and their arguments in a given document.

To this end, previous methods frequently employ a pipeline architecture including two main steps (Ji and Grishman 2008; Liao and Grishman 2010; Hong et al. 2011) as follows: (1) event classification, which involves identifying event triggers and their corresponding event types; (2) argument classification, which involves, for each detected trigger, identifying its arguments and their corresponding roles. This paper focuses primarily on the first subtask, event classification, which is important to and independent of the subsequent argument classification and strongly influences the final event extraction performance.

One difficulty in event classification is the ambiguity of the trigger words. For example, the trigger word "*beat*" in "*Obama **beat** McCain*" reflects an *Elect* event (meaning that Obama won the presidential election), but it can be easily misidentified as an *Attack* event trigger. Existing methods typically focus on exploiting sophisticated local features such as part-of-speech tags. Most of these features reflect predominantly the contextual information around given words; such features are called local features. We argue that existing local features are insufficient for disambiguation of event trigger words. Global information such as event-event association, and latent local information such as fine-grained entity types, are crucial to this task. We employ two sentences as follows to demonstrate the aforementioned issue.

*(1) He **left** the company, and he planned to **go** home directly.*
*(2) Obama **beat** McCain.*

Global Information. In the first sentence, it is difficult to tell whether "*left*" triggers a *Transport* event (meaning that a person left a place) or an *End-Position* event (a person retired from a company) when we consider only on the first clause "*He left the company*". If we could consider the texts from a broader perspective and observe that there is a *Transport* event in the second clause (triggered by *"go"*), we would have more confidence in predicting the token "*left*" to be a *Transport* event trigger because *Transport* events are more likely to co-occur with *Transport* events than with *End-Position* events. We refer to this type of global information as ***Event-event association***.

Latent Local Information. In the second sentence, if we know only that both Obama and McCain are persons, it is difficult to identify the word "*beat*" as an *Elect* event trigger because both of *Elect* and *Attack* events occur among persons and *Attack* events occur in most cases. However, if we know that both Obama and McCain are politicians, we will have ample grounds on which to predict it as an *Elect* event trigger. We refer to this type of latent information as ***Fine-grained entity types***.

Several existing methods have realized the usefulness of such information. However, they employ them only partially or mechanically. For example, Liao and Grishman (2010) proposed a two-pass ad hoc method to employ event-event association but ignored fine-grained entity types, and Hong et al. (2011) were versa; Li, Ji, and Huang (2013) proposed a jointly sentence-level approach to employ event-event association, but failed to capture it in the document level. Thus, the results of these methods are still only locally optimized values and their performance is far from satisfactory.

The above observations motivate us to simultaneously employ the aforementioned information. The key is how to

[1]This information can not be directly obtained using NLP tools, thus we call it latent information.

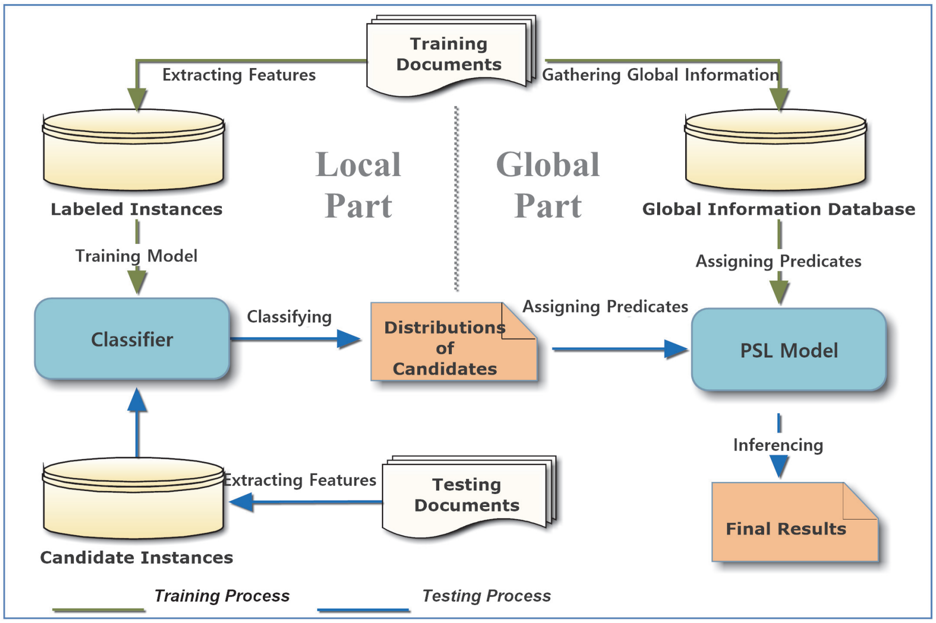

Figure 1: The framework of our approach (including training and testing processes)

encode global information. A straightforward way is to represent it as features and feed them into a classifier combined with local features. However, the biggest problem in this paradigm is that it is impossible to encode some global information (such as event-event association) as a simple feature. To resolve this problem, we propose a feasible solution which encodes global information in the form of logic. Our proposed approach consists of two parts: the local part and the global part, which focus on capturing local (including latent) and global information, respectively (see Figure 1). Specifically, (1) in the local part, we learn a classifier that employs predominantly local features to generate initial judgments for each trigger candidate; (2) in the global part, we gather "event-event" association and "topic-event" association as global information and construct a global information database; (3) we formalize both the initial judgments and the global information as first-order logic formulas and model them using Probabilistic Soft Logic (PSL) (Kimmig et al. 2012; Bach et al. 2013); and (4) finally, we generate the final results through PSL-based inference.

Note that, in our approach, local features are not modeled in the PSL. The reason is that similar to Markov Logic Networks (MLNs) (Richardson and Domingos 2006), for sophisticated local features, which are typically extremely high dimensional, it is difficult to model them using PSL (Poon and Vanderwende 2010; Venugopal et al. 2014). Thus, we use different models to capture local vs. global information. Nevertheless, through inference in PSL, all global information is captured and incorporated with a rich set of local information in a unified process. Therefore, our method is expected to achieve better performance than existing methods. It is worth noting that by virtue of the use of the first-order logic formulas, the encoded global information is intuitive for the human mind to understand and demonstrates good interpretability. Moreover, it is very convenient to incorporate new global information by adding formulas and offers high expandability.

We have conducted experimental comparisons on a widely used benchmark dataset (ACE 2005[2]). The results demonstrate that our approach is effective and achieves the best performance compared with state-of-the-art methods. In summary, our main contributions are: (1) We propose a novel approach based on PSL, which consists of the local part and the global part, to combine local (including latent) and global information for event classification. (2) We develop three types of latent features (see Section 3) which are demonstrated highly effective for event classification. (3) We explore two types of global information, event-event association and topic-event association in different textual granularity. With this global information, our proposed method achieves a considerable improvement.

Background

Task Description

The event classification task is a sub-task of the ACE evaluations. We will first introduce the ACE event extraction task. In ACE evaluations, an event is defined as a specific occurrence involving one or more participants. And event extraction task requires that certain specified types of events, which are mentioned in the source language data be detected. We introduce some ACE terminology to facilitate the understanding of this task:

Entity: an object or a set of objects in one of the semantic categories of interests.

[2] https://catalog.ldc.upenn.edu/LDC2006T06

Entity mention: a reference to an entity (typically, a noun phrase).

Event trigger: the main word that most clearly expresses an event occurrence.

Event arguments: the mentions that are involved in an event (participants).

Event mention: a phrase or sentence within which an event is described, including the trigger and arguments.

The 2005 ACE evaluation included 8 types of events, with 33 subtypes. Following previous work, we treat these simply as 33 separate event types and ignore the hierarchical structure among them. Consider the following sentence:

*He **died** in the hospital.*

An event extractor should detect a *Die* event mention, along with the trigger word "*died*", the victim "*He*" and the place "*hospital*".

Unlike the standard ACE event extraction task, we concentrate only on trigger identification and event type classification, which implies that in the previous example, our task is to identify that the token "*died*" is a trigger and that its type is *Die*.

Related Work

Event extraction is an increasingly hot and challenging research topic in NLP. Many approaches have been proposed for this task. Nearly all the existing methods on ACE event extraction use supervised paradigm. We further divide supervised approaches into feature-based methods and representation-based methods.

In feature-based methods, a diverse set of strategies has been exploited to convert classification clues (such as sequences and parse trees) into feature vectors. Ahn (2006) uses the lexical features(e.g., full word, pos tag), syntactic features (e.g., dependency features) and external-knowledge features(WordNet) to extract the event. Inspired by the hypothesis of One Sense Per Discourse (Yarowsky 1995), Ji and Grishman (2008) combined global evidence from related documents with local decisions for the event extraction. To capture more clues from the texts, Gupta and Ji (2009), Liao and Grishman (2010) and Hong et al. (2011) proposed the cross-event and cross-entity inference for the ACE event task. Li, Ji, and Huang (2013) proposed a joint model to capture the conbinational features of triggers and arguments.

In representation-based methods, candidate event mentions are represented by embedding, which typically are fed into neural networks. Two similarly related work haven been proposed on event classification (Chen et al. 2015; Nguyen and Grishman 2015). Nguyen and Grishman (2015) empolyed Convolutional Neural Networks (CNNs) to automatically extract sentence-level features for event classification. Chen et al. (2015) proposed dynamic multi-pooling operation on CNNs to capture better sentence-level features.

The Local Part

Chen and Ng (2012) proved that performing trigger identification and classification in a unified manner is superior to handling them separately. Similar to previous work, we model these activities as a word classification task. Each word in a sentence is a trigger candidate, and our objective is to classify each of these candidates into one of 34 classes (33 event types plus a NEGATIVE class). We learn a classifier to perform this task based on a set of local features. Unlike a standard classifier, the trained classifier generates a probability distribution over 34 possible labels rather than a single predicted label. We use Logistic Regression model (LR) as our classifier because of its ability in handling high-dimensional sparse features. The features presented in previous work (Ahn 2006; Li, Ji, and Huang 2013) serve as the base features. In addition, we develop several latent features.

Fine-Grained Entity Types

Hong et al. (2011) demonstrated that fine-grained entity types play an important role in event extraction. They used web information obtained from search engines to describe entity mentions. Then, they clustered all mentions based on their descriptions and treated these clusters as fine-grained entity types. However, the operation of performing online searches and extracting related information incur high time costs. Moreover, performing such searches for a large number of entity mentions is problematic because most search engines place limitations on users' query frequencies. Thus, we do not use search engines for this purpose; instead, we use WordNet to generate the descriptions of entity mentions. In detail, for a given entity mention[3], we use its related words, hypernyms and synonyms in WordNet to describe it. We perform K-means clustering algorithm based on the generating descriptions for entity mentions. Table 1 shows two clusters from our results. The labels are manually tagged.

Label	Entity Mentions
City	*New York City, New York, Chicago, Los Angeles, Rawalpindi, Bonn, Minneapolis, Basra, Mosul, San Francisco, Kirkuk, Karbala, Philadelphia, Gary, New Orleans, long beach, fort worth, hong kong, ...*
President	*George W. Bush, Lyndon Johnson, Franklin Roosevelt, truman, lincoln, tyler, kennedy, jfk, andrew jackson, bill clinton, jimmy carter, George Bush, George Washington, ...*

Table 1: Example of entity clustering results

Trigger Candidate Types

Similarly to the entity mentions, we also cluster the trigger candidates. We use the same strategy to generate the description of trigger candidates. Before clustering, we remove certain words that are unlikely to be triggers based on their part-of-speech tags. Table 2 shows examples of our results. The words in c1 tend to indicate *Attack* events, whereas the

[3]Following previous work, we use the gold standard entity mention.

Type	Words
c1	*fight, EW, assault, sortie, warfare, War, Battle, combat, battle, Defense, Combat, ASSAULT, Wars, war, siege, defense, ...*
c2	*fifty, ft, gi, en, em, twenty-three, dong, mm, seventeen, ng, lb, hm, nines, sixty, Dollar, Miles, fin, Bob, millimeter, rupee, dozen, acre, Unit, Sixers, ...*

Table 2: Example of candidates clustering results

words in c2 are numbers and units, which trigger almost no events.

Rich Context Features

As discussed in the introduction, information from a broad perspective is important for event classification. Thus, we construct several features to capture information related to the entire sentence, such as the entity types of all entities in the current sentence.

With the previous two clusters, we construct three categories of latent features as follows:

- RCF: the conjunction of Rich Context Features and base features.
- FET: the conjunction of Fine-grained Entity Types and base features.
- TCT: the conjunction of Trigger Candidate Types and base features.

The Global Part

In this part, we gather global information and incorporate it into a PSL model. Then, the inference is conducted to make the final judgment for events' classes. First, we briefly introduce PSL.

Probabilistic Soft Logic

PSL is a framework for collective, probabilistic reasoning in relational domains (Kimmig et al. 2012; Bach et al. 2013). Similar to MLNs, it uses weighted first-order logic formulas to compactly encode complex undirected probabilistic graphical models. However, PSL brings two remarkable advantages compared to MLNs. First, PSL relaxes the boolean truth values of MLNs to continuous, soft truth values. This allows for easy integration of continuous values, such as similarity scores. Second, PSL restricts the syntax of first order formulas to that of rules with conjunctive bodies. Together with the soft truth values constraint, the inference in PSL is a convex optimization problem in continuous space and thus can be solved using efficient inference approaches. For further details, see the references (Kimmig et al. 2012; Bach et al. 2013).

Encoding Global Information

There are two types of global information that we wish to incorporate into our method: the event-event association and topic-event association.

Event-Event Association The probability of events' co-occurrence is closely related to their types. For example, an *Attack* event is much more likely to co-occur with a *Die* event than with a *Marriage* event. We use the conditional probability $p(t_1 \mid t_2)$, which denotes the probability of observing a t_1 type event given that a t_2 type event has been observed, to represent the event-event association. We calculate this probability at both the sentence and document levels, which are denoted by p_{sen} and p_{doc}, respectively.

$$p_{sen}(t_1 \mid t_2) = \frac{num_{sen}(t_1, t_2)}{\sum_{t \in T} num_{sen}(t, t_2)} \quad (1)$$

$$p_{doc}(t_1 \mid t_2) = \frac{num_{doc}(t_1, t_2)}{\sum_{t \in T} num_{doc}(t, t_2)} \quad (2)$$

In Equation 1, t_1 and t_2 denote event types; $num_{sen}(t_1, t_2)$ is the co-occurrence frequency between t_1 type events and t_2 type events in the same sentence; and T is the set of all possible event types. The meanings of symbols in Equation 2 are similar but apply at the document level.

We define two indicator functions $I_{sen}(c_1, c_2)$ and $I_{doc}(c_1, c_2)$, where the symbols c_1, c_2 denote trigger candidates. $I_{sen}(c_1, c_2)$ is true when c_1 and c_2 are in the same sentence. $I_{doc}(c_1, c_2)$ is similar, but applies at document level. Finally, we define four predicates to encode this information in the PSL model, which are listed in the upper portion of Table 3. The symbols t_1 and t_2 denote event types.

Topic-Event Association A document on a certain topic tends to describe events of several certain types. For example, entertainment news items often describe events of *Marriage* and *Born* types but almost never include events of *Attack* or *Die* types. We apply a Latent Dirichlet Allocation (LDA) (Blei, Ng, and Jordan 2003) model to the ACE corpus and label each document with a topic. Then, we calculate the probability of observing an event of type t in a document on topic p,

$$p_t(t \mid p) = \frac{num(t, p)}{\sum_{t' \in T} num(t', p)} \quad (3)$$

Figure 2 shows the distribution of *Attack*, *Transport* and *Die* events among the considered topics. As shown in the figure, the topic of a document is a strong indicator of the events that it contains. For example, a document on topic 14 is more likely to contain *Attack* events than *Transport* events.

We define a indicator function $I_t(c, p)$, which is true when the topic of the document containing c is p. Finally, we define two predicates listed in the middle portion of Table 3 to encode this information in our method.

Inference

Beltagy, Erk, and Mooney (2014) found that the standard formula for conjunction in PSL is overly restrictive and does not work well for semantic textual similarity. The same issue arises in our task. Therefore, following them, we redefine the formula for conjunction as $I(l_1 \wedge l_2 \wedge \cdots \wedge l_n) = \frac{1}{n} \sum_{i=1}^{n} I(l_i)$.

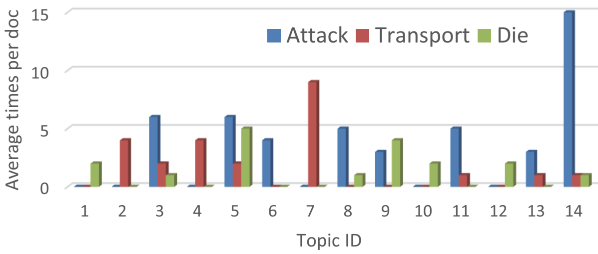

Figure 2: Topic-event distribution

Type	Predicate	Assignment
Event-Event	$sameSen(c_1, c_2)$	$I_{sen}(c_1, c_2)$
	$senLevel(t_1, t_2)$	$p_{sen}(t_1 \mid t_2)$
	$sameDoc(c_1, c_2)$	$I_{doc}(c_1, c_2)$
	$docLevel(t_1, t_2)$	$p_{doc}(t_1 \mid t_2)$
Topic-Event	$topic(c, p)$	$I_t(c, p)$
	$topicEvt(p, t)$	$p_t(t \mid p)$
Local-Part	$candEvt(c, t)$	$p_l(c, t)$

Table 3: Observed predicates and their corresponding assignments

We define the predicate $eventType(c, t)$ to indicate that candidate c triggers an event of type t. It is the only target predicate in our model, whose assignments are not given during inference and thus need to be predicted. All others are observed predicates, which are always assumed to be known during inference. Table 3 lists all the observed predicates and their corresponding assignments, where $candEvt(c, t)$ is used to encode the probability $p_l(c, t)$, which is the initial judgments generated by the local part. Putting all predicates together, we design three formulas to apply the aforementioned information in PSL (see Table 4). Formula $f1$ encodes the relationship between document topics and event types (topic-event association). Formulas $f2$ and $f3$ model the relationship between event types in sentence and document level, respectively (event-event association).

$f1$	$topic(c, p) \wedge topicEvt(t, p) \wedge candEvt(c, t) \rightarrow eventType(c, t)$
$f2$	$sameSen(c_1, c_2) \wedge senLevel(t_1, t_2) \wedge candEvt(c_2, t_2) \wedge eventType(c_1, t_1) \rightarrow eventType(c_2, t_2)$
$f3$	$sameDoc(c_1, c_2) \wedge docLevel(t_1, t_2) \wedge candEvt(c_2, t_2) \wedge eventType(c_1, t_1) \rightarrow eventType(c_2, t_2)$

Table 4: Formulas in the PSL model

We manually set the formulas' weights in our experiments[4]. The inference results provide us with the most likely interpretation, that is, the soft-truth assignments to the predicate $eventType$. By choosing a threshold for the truth values in the interpretation, we can select a set of grounded atoms of the target predicate with high confidence.

[4]Manually setting the weights is a common strategy for PSL when there is a lack of sufficient training data (Pujara et al. 2013; Memory et al. 2012). The weights of f1, f2 and f3 were set to 10, 2 and 1, respectively

Experiments

Data Set and Experimental Setup

We performed experiments on the ACE 2005 corpus. For the purpose of comparison, we followed the evaluation of Li, Ji, and Huang (2013): randomly selected 30 articles from different genres as the development set, and we subsequently conducted a blind test on a separate set of 40 ACE 2005 newswire documents. We used the remaining 529 articles as our training data set. The corpus was processed using the Stanford CoreNLP Toolkit (Manning et al. 2014).

Following previous work (Liao and Grishman 2010; Hong et al. 2011; Li, Ji, and Huang 2013), we use the following criteria to evaluate the results:

1. A trigger is correctly identified if its offset matches a reference trigger.
2. A trigger is correctly classified if both its event type and offset match a reference trigger.

We searched for the hyperparameters on the development set. We searched for both the number of fine-grained entity clusters and the number of trigger clusters (denoted by n_{fec} and n_{ftc}, respectively) in $\{50k \mid k = 1, 2, \cdots, 10\}$, the number of topics n_p in $\{10k \mid k = 1, 2, \cdots, 10\}$, and the truth-value threshold thr in $\{0.05k \mid k = 1, 2, \cdots, 20\}$. These parameters were independently sought. The selected values used in our experiment were $n_{fec} = 200$, $n_{ftc} = 300$, $n_p = 50$ and $thr = 0.6$.

Overall Performance

Table 5 shows the experimental results obtained on the blind test set. Since most previous work did not report identifying results, we only compare classifying performances to state-of-the-art methods in this part, leaving identifying performances in other parts of this section. From the results, we can state the following observations.

1. *LR(base+latent)* outperforms *LR(base)* significantly with a gain of 2.1% improvement, which demonstrates that the latent features proposed in Section 3 are highly effective for this task.
2. Compared with *LR(base+latent)*, with incorporating of global information, *Combined PSL* achieves a gain of 1.4% improvement, thereby demonstrating that the global information is important to this task. To sum up, combining the latent and global information enables our proposed approach to obtain a gain of 3.5% improvement in total.
3. *Nguyen's CNN* and *Chen's DMCNN* are the latest work on this task, which achieved the best performance among state-of-the-art methods. Our method outperforms both of them and further demonstrates the effectiveness of our approach.

Effects of Latent Features

We investigate the effects of the latent features proposed in Section 3 both to classifiers and to our proposed model. Table 6 shows the results (F_1 value). It is evident that RCF

Methods	Pre	Rec	F1
Li's baseline	74.5	59.1	65.9
Liao's cross-event	68.7	68.9	68.8
Hong's cross-entity	72.9	64.3	68.3
Li's joint model	73.7	62.3	67.5
Nguyen's CNN	71.8	66.4	69.0
Chen's DMCNN	75.6	63.6	69.1
LR(base)	69.6	62.5	65.9
LR(base+latent)	75.2	62.1	68.0
Combined PSL	75.3	64.4	**69.4**

Table 5: Overall performances. Both *LR(base)* and *LR(base+latent)* are logistic regression models. The former model uses the base features only, and the latter model additionally uses latent features. *Combined PSL* is our proposed approach, which uses both the latent and global information.

features enable the classifier to achieve more improvements for classification than for identification (0.9% vs. 0.1%), and TCT features are versa (0.6% vs. 1.3%). The previous observation also persists in the combined PSL model. We believe this phenomenon occurs because of the follows reasons: RCF features reflect rich information around a given candidate, and this specific information can indicate its event type; whereas, TCT features are obtained from clustering results which reflect the high-level and coarse information of the candidate, thus they could only indicate the coarse event type (i.e. whether it is a trigger or not) but are not specific enough to indicate the event type. In addition, the results also show that, with incorporating all these features, both the classifier and our proposed approach achieve considerable improvements (more than 1.5%), which demonstrate that these latent features are highly effective for this task.

Features	Classifier		Combined PSL	
	Ident	Class	Ident	Class
Base Features	68.3	65.9	69.7	67.9
+ RCF	68.4	66.8	69.8	68.6
+ FET	69.2	67.1	70.6	68.9
+ TCT	69.6	66.5	71.1	68.2
+ all	**70.7**	**68.0**	**71.7**	**69.4**

Table 6: Effects of latent features. RCF, FET, and TCT represent our three categories of latent features(see Section 3.3). The column labeled with "Classifier" shows the results of classifiers with different features; the column labeled with "Combined PSL" shows the results of our proposed models with different features in the local part.

Effects of Different Types of Global Information

Our proposed approach uses two types of global information: the event-event association and the topic-event association. Their detail effects are illustrated in Table 7, where "No Global Information (NGI)" indicates that the truth values of $eventType(c,t)$ were directly set to $candEvt(c,t)$[5]. From the table, we observe that both types of global information help the proposed model to obtain better performances. We achieve the best performance when incorporating both of them simultaneously.

Features	Ident (%)	Class (%)
	F1	F1
No Global Information	70.6	68.1
+ event-event	70.8	68.7
+ topic-event	71.2	69.0
+ all	**71.7**	**69.4**

Table 7: Effects of global information

Discussion

Topic-event association reflects document level information, thus we consider it as global information and apply it in the global part in our experiments. However, unlike event-event association, technically, it could also be incorporated in the local part. We construct two systems, one encodes the topic-event association in local part and the other encodes that in global part, to investigate the effects. We observe that employing this information in the local part yields less improvement (68.8%/69.4%). This may occur because when this global information is combined with the high-dimensional local features, the local part is unable to effectively learn this single feature, causing its effect to be weakened. This finding may indicate that it is necessary to treat the high-dimensional local features and the global features separately. However, this topic is out of the scope of this work, we will defer it to a future study.

Conclusions

We propose a novel approach based on PSL, which consists of the local part and the global part, to exploit local (including latent) and global information in event classification. In the local part, we develop several latent features which are demonstrated highly effective for event classification. In the global part, we explore two types of global information, event-event association and topic-event association, in different textual granularity. The experimental results show that, with incorporating the latent and global information, our proposed approach obtains a gain of 3.5% improvement in total. Moreover, we outperforms state-of-the-art methods and achieve the best performance on the benchmark dataset.

Acknowledgements

The work was supported by the Natural Science Foundation of China (No. 61533018), the National Basic Research Program of China (No. 2014CB340503) and the National Natural Science Foundation of China (No. 61272332).

[5]NGI differs from LR(base+latent). The former is a Combined PSL model, whereas the latter is a classifier.

References

Ahn, D. 2006. The stages of event extraction. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, 1–8.

Bach, S. H.; Huang, B.; London, B.; and Getoor, L. 2013. Hinge-loss Markov random fields: Convex inference for structured prediction. In *Proceedings of Uncertainty in Artificial Intelligence (UAI)*.

Beltagy, I.; Erk, K.; and Mooney, R. 2014. Probabilistic soft logic for semantic textual similarity. In *Proceedings of ACL*, 1210–1219.

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *the Journal of Machine Learning Research* 993–1022.

Chen, C., and Ng, V. 2012. Joint modeling for chinese event extraction with rich linguistic features. In *COLING*, 529–544.

Chen, Y.; Xu, L.; Liu, K.; Zeng, D.; and Zhao, J. 2015. Event extraction via dynamic multi-pooling convolutional neural networks. 167–176. Association for Computational Linguistics.

Gupta, P., and Ji, H. 2009. Predicting unknown time arguments based on cross-event propagation. In *Proceedings of ACL-IJCNLP*, 369–372.

Hong, Y.; Zhang, J.; Ma, B.; Yao, J.; Zhou, G.; and Zhu, Q. 2011. Using cross-entity inference to improve event extraction. In *Proceedings of ACL*, 1127–1136.

Ji, H., and Grishman, R. 2008. Refining event extraction through cross-document inference. In *Proceedings of ACL*, 254–262.

Kimmig, A.; Bach, S.; Broecheler, M.; Huang, B.; and Getoor, L. 2012. A short introduction to probabilistic soft logic. In *Proceedings of NIPS Workshop*, 1–4.

Li, Q.; Ji, H.; and Huang, L. 2013. Joint event extraction via structured prediction with global features. In *Proceedings of ACL*, 73–82.

Liao, S., and Grishman, R. 2010. Using document level cross-event inference to improve event extraction. In *Proceedings of ACL*, 789–797.

Manning, C. D.; Surdeanu, M.; Bauer, J.; Finkel, J.; Bethard, S. J.; and McClosky, D. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of ACL*, 55–60.

Memory, A.; Kimmig, A.; Bach, S.; Raschid, L.; and Getoor, L. 2012. Graph summarization in annotated data using probabilistic soft logic. In *Proceedings of the 8th International Workshop on Uncertainty Reasoning for the Semantic Web (URSW 2012)*, volume 900, 75–86.

Nguyen, T. H., and Grishman, R. 2015. Event detection and domain adaptation with convolutional neural networks. Association for Computational Linguistics.

Poon, H., and Vanderwende, L. 2010. Joint inference for knowledge extraction from biomedical literature. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 813–821. Association for Computational Linguistics.

Pujara, J.; Miao, H.; Getoor, L.; and Cohen, W. 2013. Knowledge graph identification. In *The Semantic Web–ISWC 2013*. Springer. 542–557.

Richardson, M., and Domingos, P. 2006. Markov logic networks. *Machine learning* 107–136.

Venugopal, D.; Chen, C.; Gogate, V.; and Ng, V. 2014. Relieving the computational bottleneck: Joint inference for event extraction with high-dimensional features. In *Proceedings of EMNLP*, 831–843.

Yarowsky, D. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of ACL*, 189–196.